

# L<sup>A</sup>T<sub>E</sub>X 201

Kód szerkezete, szintaktika – mélyebben

Vadon Viktória

2022/23/I. félév

# Parancsok

## Emlékeztető

parancsok egy típusa \+ betűsorozat

- csak az angol abécé kis- és nagybetűi

miért?

- a kód feldolgozása során addig olvas, amíg speciális karaktert, számot vagy whitespace-t (szóköz, enter, stb.) nem talál

következésképp:

- a parancs nem tartalmazhat számot, speciális karaktert, whitespace-t
- ezért nem folyhat össze a következő szóval sem!

# Parancsok és környezetek argumentumai

- általában:

```
\parancs [opc1, opc2] {köt1}{köt2}
```

- [ ]: opcionális argumentum(ok), közös szögletes zárójelben, vesszővel elválasztva
  - { }: kötelező argumentum(ok), külön-külön kapcsos zárójelekben
- környezetre hasonlóképp, `\begin{környezetnév}` után:

```
\begin{környezetnév} [opc1, opc2] {köt1}{köt2}
```

# Argumentumok és whitespace I

## Emlékeztető

*általában* nem szereti az argumentumok *között* a szóközöket és sortöréseket

- kommenteléssel lehet becsapni

több opcionális argumentum közös szögletes zárójelbe [], vesszővel elválasztott lista

- *sokszor* az opcionális argumentumok listájában sem szereti a szóközt
- miért?
  - a *preambulumban* feldolgozási hibát okoz, zárójelek és vesszők mentén darabol, ha whitespace-t lát, azt hiszi, vége a parancsnak
  - a *dokumentumtörzsben* kevésbé kényes a whitespace-re!

# Argumentumok és whitespace II

## Argumentumon *belül*

- *sokszor* a kötelező argumentumon belül *szóköz* megengedett, *sortörés* *sokszor* nem
  - ez igaz pl. formázási parancsokra (kiemelés, vastag betű, stb.)
- *környezet* esetén
  - a környezet *argumentumaira* hasonló szabályok érvényesek
  - a környezet *törzsében* általában megengedett sortörés/új paragrafus is!

# Csomagok – újra

## Emlékeztető

csomag betöltése beállításokkal együtt

```
\usepackage[opció1,opció2]{csomagnév}
```

- bizonyos beállítások `kulcs = érték` formátumban adandók meg, de ugyanúgy listában:

```
\usepackage[opció,kulcs1=érték1,kulcs2=érték2]%
{csomagnév}
```


- több csomag is betölthető egyszerre(!):<sup>1</sup>

```
\usepackage{csomag1,csomag2}
```

- ha egyikhez sem kellenek beállítások
- vagy éppen *közösek* a beállítások

```
\usepackage[beállítások]{csomag1,csomag2}
```

---

<sup>1</sup>Kivételes eset, amikor a kötelező argumentumban lehet `lista` 

# Parancsok, környezetek, csoportok egymásba ágyazása I

- *általában* egymásba ágyazhatók, pl.

```
\textit{Dőltön belül \textbf{vastagbetűs} szöveg.}
```

*Dőltön belül **vastagbetűs** szöveg.*

- *bizonyos* parancsok csak adott környezeten belül léteznek!
  - pl. listáknál a listaelem `\item` parancsa csak a lista környezeten (`itemize`, `enumerate`) belül működik
- *bizonyos* „törékeny” parancsok nem ágyazhatók be
  - pl. a *verbatim* környezet: feldolgozás nélkül, karakterről karakterre bemásolt szöveg. (célja formázás megtartása, vagy forráskód beszúrása)

## Parancsok, környezetek, csoportok egymásba ágyazása II

- elég mélyen lehet egymásba ágyazni
- több alcsoport is lehetséges egy csoporton belül, pl.
 

```
{csoport \small kicsi {\it dőlt} kicsi
{\bf vastag} kicsi}
csoport kicsi dőlt kicsi vastag kicsi
```
- csak arra vigyázzunk, ne vesszünk el a zárójelek között
- de! nem lehet „átfedés”, mindig a legmélyebb szintű csoport/környezet kell, hogy először lezáruljon!
  - pl. B környezetben C csoport/környezet, C-t be kell zárni B előtt
  - (névtelen) csoportok esetén automatikusan úgy értelmezi, hogy a legmélyebb szintet zárjuk be



# Furcsaságok, amiket azért jó tudni I

## Dobozok és ragasztó

- a L<sup>A</sup>T<sub>E</sub>X (és T<sub>E</sub>X) fejében minden egymásba ágyazott dobozokból épül fel: karakterek, szavak, mondatok, bekezdések, ábrák, hasábok, oldalak, stb.
- köztük glue = ragasztó, ami szükség esetén nyújtható vagy zsugorítható, pl. változó betű- és szóközök a sorkizártsághoz, paragrafusok közti távolság változik a függőleges helykitöltéshez, stb.
- a program a tipográfus, többnyire hagyjuk ezt rá
- ha magunk definiálunk méreteket valahol, mindig adjuk meg, hogy mennyivel nyújtható és zsugorítható

## Furcsaságok, amiket azért jó tudni II

### Whitespace a kódban és a lefordított dokumentumban

- a kódban elhelyezett whitespace, mint soremelések, tabulátorok, szóközök, *nem egy az egyben* mennek át a dokumentumba!
- két enter (avagy egy üres sor) a kódban = új paragrafus a dokumentumban
- szinte minden más whitespace kombináció = egy szóköz
- pl. a kódban elhelyezett tabulátorok és behúzások nem jelennek meg a dokumentumban!
  - alapvetően bízzuk a T<sub>E</sub>X motorra
  - később tanulunk majd rá módot, hogy hogyan lehet egy az egyben átvinni a formázást, pl. ha képvershez muszáj
- pozitívum: (szinte) tetszés szerint tagolhatjuk és formázhatjuk a kódot, hogy nekünk áttekinthető legyen!

# L<sup>A</sup>T<sub>E</sub>X források

## Magyarul

- röviden: Gyimesi Eszer könyve
- hosszabban: L<sup>A</sup>T<sub>E</sub>X 78 percben
- hosszan: Tómacs Tibor könyve

## Angolul

- L<sup>A</sup>T<sub>E</sub>X wikibooks
- parancssorból `texdoc latex2e` (quick reference guide),  
`texdoc source2e` (commentated source code)
- csomagok dokumentációi
- kérdések: Google > StackOverflow
- online reference manuals
- stb.

# T<sub>E</sub>X források

Hasznos lehet, mert a T<sub>E</sub>X alacsonyabb szintű parancsai is használhatók L<sup>A</sup>T<sub>E</sub>X-ben – bár sokra van egyszerűbb, átláthatóbb, elegánsabb L<sup>A</sup>T<sub>E</sub>X megoldás.

Angolul tudóknak

- Donald E. Knuth: The T<sub>E</sub>Xbook. Plain T<sub>E</sub>X forráskód<sup>2</sup> (ahogy Knuth maga megosztotta) kész pdf (az internetről)
- Paul W. Abrahams, Kathryn A. Hargreaves, and Karl Berry: T<sub>E</sub>X for the Impatient

Magyarul

- Bujdosó Gyöngyi, Fazekas Attila: T<sub>E</sub>X kezdőlépések

a diasor összeállításához használt egyéb források:

- L<sup>A</sup>T<sub>E</sub>X kézikönyv ingyenes részlete
- Dr. Olajos Péter jegyzete

---

<sup>2</sup>MikT<sub>E</sub>X-kel érkező T<sub>E</sub>XWorks programmal pdftex motorral fordítható